

一個可減少危險區域的後置通道繞線處理器

程仲勝

大葉大學電機工程學系
彰化縣大村鄉山腳路 112 號

摘要

隨著積體電路中線路之間的距離越來越小且整個電路的運算速度也越來越快，因此交感雜訊 (crosstalk noises) 和瑕疵點 (spot defects) 的問題亦日趨重要。交感雜訊是由於相鄰線路間相互電容及電感作用之結果。其可增長訊號延遲時間亦能導致不預期之邏輯轉變。另一方面，在積體電路製造過程中，若額外多餘物質之附著或部份製造物質之脫落，相鄰線路間便會產生瑕疵點，此時恐將導致線路之間短路或斷路。在積體電路的設計和製造過程中，交感雜訊和瑕疵點將會大大降低晶片效能和產量。因此對超大型積體電路實體設計 (physical design) 後段之詳細繞線 (detailed routing) 階段而言，如何減少相鄰線路之間易產生交感雜訊和不良瑕疵點的危險區域 (critical regions) 已成為一相當重要的課題。

本論文中我們將考慮解決通道繞線 (channel routing) 危險區域最小化的問題。由於直接設計一具有最少危險區域的通道繞線演算法之問題複雜度相當高，因此取而代之的是去改進現存通道繞線演算法的結果以減少危險區域。所提出的後置繞線處理器採用軌道重新排列 (track permutation) 和佈局層重新指定 (layer assignment) 兩階段以改進繞線結果。在軌道重新排列階段，我們採用一個以圖形為主的啟發式演算法將所有繞線軌道重新排列。至於佈局層重新指定方面，我們針對繞線線段建立有權重的平面圖，並求其具有最大權重之獨立集 (maximum weighted independent set)，以解決繞線線段佈局層重新指定問題。

最後，我們以一些例子來測試評估所提出之能減少危險區域之後置通道繞線處理器，實驗結果證實所提方法確實能有效減少危險區域的數量。

關鍵詞：交感雜訊，瑕疵點，危險區域，積體電路實體設計，通道繞線

A Postprocessing Channel Router for Critical Region Minimization

JONG-SHENG CHERNG

*Department of Electrical Engineering, Da-Yeh University
112 Shan-Jiau Rd., Da-Tsuen, Changhua, Taiwan*

ABSTRACT

As interconnection wire spacing becomes smaller and the circuit operating frequency runs higher, the problems of crosstalk noises and spot defects become very crucial. Since crosstalks are a

result of mutual capacitances and inductances between neighboring wires, they can cause increased delays or inadvertent logic transitions. Spot defects are caused by extra or missing material during the manufacturing process. They may cause a short or an open circuit between neighboring wires. During integrated circuit design and manufacturing, crosstalks and spot defects will greatly reduce the performance and the yield of chips. Consequently, reduction of the critical regions between interconnection wires resulting from crosstalks and spot defects becomes an important consideration in the detailed routing phase of VLSI physical design.

In this paper, the critical region minimization problem for channel routing is considered. Due to the high complexity of directly developing a critical region minimum channel routing algorithm, a postprocessor to improve the results of existing routing algorithms by minimizing critical regions is proposed instead. In this postprocessor, two phases, namely track permutation and layer assignment, are considered. In the track permutation phase, a graph-based heuristic is proposed for permuting the routing tracks. As for the layer assignment phase, some weighted planar graphs can be derived from the routing segments; then, the maximum weighted independent set for each can be found so as to obtain a critical region minimization layer assignment solution.

The proposed postprocessor was evaluated with some channel routing examples. The experimental results showed that the number of critical regions between adjacent wires for each example is minimized.

Key Words: crosstalk noises, spot defects, critical regions, VLSI physical design, channel routing

一、簡介

隨著超大型積體電路設計及製造技術的進步，電路元件間放置的距離日漸縮短，再者隨著線路及電晶體交換延遲時間的縮短，因而導致快速的訊號轉換，所有上述因素都會增加線路元件間的耦合電容。然而耦合電容的增加不僅會導致較長的訊號傳輸時間，並且由於交感雜訊（crosstalk noises）的影響而產生訊號傳輸的不一致性，進而導致整個系統運作發生錯誤。另一方面，由於電路元件及線路擺放距離日漸縮短，相鄰線路及元件間較易由於額外多餘物質之附著或部份製造物質之脫落，而產生導致線路之間短路或斷路之瑕疵點（spot defects）。在積體電路的設計和製造過程中，交感雜訊和瑕疵點將會大大地提高測試成本及降低系統效能和產量 [2]，因此如何減少線路元件間易產生交感雜訊和不良瑕疵點的危險區域（critical regions）已成為一個相當重要的研究課題。

基本上發展一具有最少危險區域的繞線演算法比一般傳統繞線演算法更加困難，這是因為具有最少危險區域的演算法除了必須決定每一個別的繞線位置以外，同時亦需考慮線段間的相對位置，以避免產生過大的交感雜訊以及過多的不良瑕疵點。

一般而言，具有此類效能驅動繞線問題可分為兩類繞線

模式來解決。第一種是使用無格線繞線（gridless routing）模式，藉由調整格線空間大小以降低交感雜訊及減少不良瑕疵點的數目，達到具有最少危險區域的效果。如在文獻 [3,10] 中，作者利用無格線繞線模式來降低交感雜訊，但其並未具體考慮瑕疵點的問題。第二種為使用有格線繞線（gridded routing）模式，此種模式中的每一繞線線段必須放置在固定寬度的格線內，無法調整格線空間大小，因此必須在繞線時考慮線段間的相對位置以減少危險區域。目前此類有格線繞線模式的作法只分別應用於減少交感雜訊或減少易形成不良瑕疵點的危險區域，而並未同時考慮降低兩者所形成的危險區域數目。單純降低交感雜訊的作法，已應用在平面繞線 [4,5]，通道繞線 [1,6,8,11] 及四面通道繞線 [7]。而在文獻 [9] 中，亦只單純減少易產生瑕疵點的危險區域而完全忽略交感雜訊所造成的影響。

本論文中，我們將處理同時減少交感雜訊及瑕疵點所形成危險區域的有格線通道繞線（channel routing）問題。由於在繞線階段時以全區域性的角度來考慮危險區域最少化問題是一件耗時且相當困難的問題，因此我們的作法並不會直接尋求一個啟發式的繞線演算法，而是去修改一個現存有格線繞線演算法的結果以減少危險區域的數目。此種作法的好處之一是能善加利用現存繞線演算法的優點，並將問題的

複雜度降低。在此論文中，我們將提出一個後置通道繞線處理器，它首先接受一有格線初始繞線結果，然後重新排列軌道的順序，接著再重新指定目前繞線線段的位置，以減少網列 (nets) 間危險區域的數目。

本論文往後章節組織如下：第二節為欲解決問題之描述。第三節詳細介紹我們提出來解決危險區域問題之後置通道繞線演算法。第四節為實驗結果，實驗內容顯示我們提出的通道繞線器確實可減少繞線連線組間危險區域的數目。最後第五節為結論。

二、問題描述

繞線是 VLSI 實體設計不可或缺的一環，而有格線通道繞線問題 (gridded channel routing problem) 則為詳細繞線的一種特殊型態問題。它最主要的目的是在滿足一些限制條件下將位於通道上下兩邊的接腳 (pin) 集合在格子化通道繞線區域內相互連接而使得通道高度達到最小，如圖 1 所示。但隨著通道中連接線間相互距離越來越小，使得容易產生交感雜訊和瑕疵點的危險區域越來越多 (如圖 1 灰色區域所示)，進而影響繞線品質。因此我們的目標為發展一可得到最少危險區域的繞線演算法。

在著手進行繞線演算法設計之前，必須先分析與評估任兩線段之間交感雜訊的大小以及其間產生瑕疵點的機率。首先我們將評估交感雜訊的大小，交感雜訊為相鄰連接線間寄生耦合現象而導致的電磁干擾現象 [2]。一般而言，相鄰兩連接線間交感雜訊值的大小與其耦合平行長度成正比，而與其距離成反比 [2]。然而由於兩平行線段間的耦合電容會隨著其間距離的增加而快速遞減，因此為了簡化問題起見，在本論文中我們可合理的假設交感雜訊只存在於格子化繞線區域之相鄰列 (row) 或相鄰行 (column) 間，如圖 1 方形灰色區域所示。

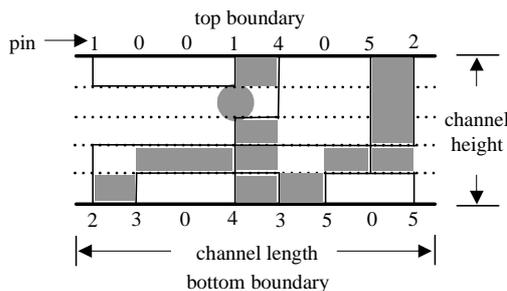


圖 1. 有格線通道繞線

另一方面，我們亦需評估兩相鄰線段所圍成區域，其間產生瑕疵點的可能性。在圖 1 中所有灰色區域為相鄰線路間可能由於額外多餘物質之附著或部份製造物質之脫落而產生導致線路之間短路或斷路之瑕疵點所在區域。在此我們亦假設非相鄰行與非相鄰列間瑕疵點存在的機率為零。然而相鄰線路間形成瑕疵點的機率不同，一般認為鄰近貫孔 (via) 所形成瑕疵點的機率大於其他地方。綜上所述，減少交感雜訊及瑕疵點所形成危險區域的有格線通道繞線問題即可轉換成減少所有相鄰行與相鄰列間由於不同繞線網列所圍成危險區域的有格線通道繞線問題。如圖 1 之通道繞線結果可知危險區域的個數為 14。本論文所欲解決的問題為針對一組網列在通道內百分之百完成連線並使得危險區域的個數為最少。以圖 1 之網列而言，在同樣利用四條軌道 (track) 完成繞線的情況下，危險區域的個數可比 14 更少。

三、後置通道繞線演算法

為了降低問題的複雜度，我們不直接設計具有最少危險區域的有格線繞線演算法，而是修改一現存繞線演算法的結果以減少危險區域的數目。在我們目前所採用的繞線模式中，有兩層佈局層可供繞線。以下即分別說明所提出之後置通道繞線演算法 (postprocessing channel routing algorithm) 中所採取的兩個階段步驟：

(一) 軌道重新排列 (track permutation)

首先我們將利用一現存之繞線演算法得到一初始繞線結果，再針對此一初始繞線結果，將其水平軌道重新排列以減少危險區域數目。假設 S 為一初始繞線結果，從上邊界到下邊界其使用了 d 個軌道 t_1, t_2, \dots, t_d 完成繞線。讓 P 代表軌道 t_1, t_2, \dots, t_d 的一種重新排列。我們知道並非每一個任意的軌道重新排列都是合理的。很明顯地， P 是一個合理的軌道重排若且惟若重排後的繞線結果 $P(S)$ 沒有來自不同網列線段的垂直重疊 (vertical overlap)，亦即需符合垂直限制 (vertical constraint) 之條件。為了描繪合理軌道重排的特性，我們針對初始繞線解 S 定義軌道次序圖 (track ordering graph) $G(S)$ 。在 $G(S)$ 圖中的每個節點代表一個軌道，而圖中一個從軌道 t_i 指向軌道 t_j 的有方向性邊 (directed edge) 則意味著軌道 t_i 必須擺放在軌道 t_j 之上。現在，問題的目標是去獲得一合理軌道重排 P 如此以至於在 $P(S)$ 中危險區域的數目為最少。目前我們採用以下啟發式經驗法則來解決軌道重新排列的問題：

步驟一：輸入一個兩層的初始繞線結果 S 。

步驟二：建構軌道次序圖 $G(S)$ 。

步驟三：交替重複以下子步驟 a 及 b 直到 $G(S)$ 的頂點被完全移除。

- a. 找出在 $G(S)$ 中沒有被箭頭指到的頂點集合，接著從此集合中挑出一個與目前重排結果產生最少危險區域的頂點 v ，並將其擺放在軌道位置 t_i (i 從 1 到 $d/2$)。最後，從 $G(S)$ 中移除頂點 v 和其所伴隨的所有有向邊。
- b. 找出在 $G(S)$ 中沒有箭頭指出的頂點集合，接著從此集合中挑出一個與目前重排結果產生最少危險區域的頂點 v ，並將其擺放在軌道位置 t_j (j 從 d 到 $d/2$)。最後，從 $G(S)$ 中移除頂點 v 和其所伴隨的所有有向邊。

假設圖 1 為初始繞線結果 S ，圖 2 為其軌道次序圖 $G(S)$ 。在執行完上述軌道重新排列演算法後，其最終繞線結果顯示在圖 3。從圖 3 中可知，原軌道 t_1 及 t_4 位置不變，但軌道 t_2 及 t_3 位置已交換，而新的繞線結果所產生的危險區域數目已從原來的 14 降至 11。整個軌道重新排列演算法以步驟二及步驟三最為耗時，其時間複雜度為 $O(d^2)$ ，其中 d 為軌道之數目。

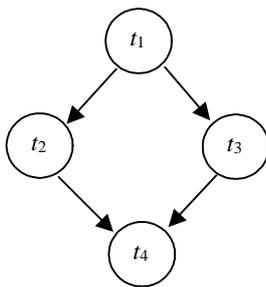


圖 2. 軌道次序圖 $G(S)$

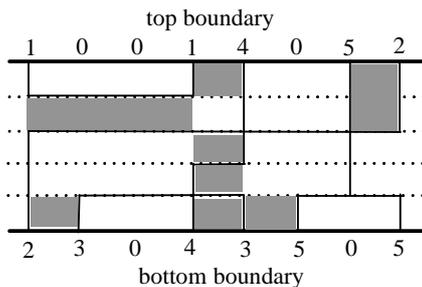


圖 3. 軌道重排後之通道繞線結果

(二) 佈局層重新指定 (layer assignment)

大部分的通道繞線演算法均假設每一繞線層的繞線方向固定，亦即非水平方向繞線則為垂直方向繞線。但在我們佈局層重新指定的階段裡，此種限制將被解除，因此網列線段可被重新指定擺放至佈局層中以減少危險區域的數目。在完成軌道重新排列後，我們將得到一新的繞線結果。針對此一繞線結果，將其水平與垂直線段作一佈局層重新指定，以求得一具有更少危險區域的繞線結果。以下為佈局層重新指定所採取之步驟：

步驟一：首先將繞線結果切割成幾個獨立的群聚 (clusters)。

每個群聚構成一最大局部佈局 (maximal partial layout)，亦即在每個群聚裏，任一網列線段之佈局層重新指定皆會直接或間接影響到所在群聚中其他網列線段所形成危險區域之消長。透過此步驟我們可將原來整個通道佈局層重新指定問題切割成數個較小的佈局層重新指定子問題，藉此可簡化問題的複雜度。以圖 3 之軌道重排後之通道繞線結果為例，步驟一可將其切割成一個獨立的群聚。此群聚包含所有的網列線段，因為網列一的水平線段佈局層重新指定會影響到與網列二的水平線段所形成危險區域的數量，而網列二的水平線段佈局層重新指定會影響到與網列四的水平線段所形成危險區域的數量，同理網列四會影響到網列三，最後網列三的垂直線段佈局層重新指定會影響到與網列五的垂直線段所形成危險區域的數量。

步驟二：針對每一群聚再切割成一水平子群聚 (H-cluster)

及一垂直子群聚 (V-cluster)。以圖 3 之軌道重排後之通道繞線結果為例，其水平子群聚與垂直子群聚分別如圖 4(a) 及 4(b) 所示。

步驟三：針對每一子群聚，我們將建立一個有權重的平面圖

(weighted planar graph) $G(V, E)$ 。在 $G(V, E)$ 中的每個頂點代表一條線段，而連接某兩頂點的無向邊意味著此兩頂點 (此兩線段) 之間可形成危險區域；每一頂點的權重為其本身與其他相鄰頂點可產生危險區域數目之和。以圖 4(a) 及 4(b) 之水平子群聚與垂直子群聚為例，其有權重的平面圖分別如圖 5(a) 及 5(b) 所示。

步驟四：最後，在每個子群聚所對應之權重平面圖中，尋求一具有最大權重之獨立集 (maximum weighted in-

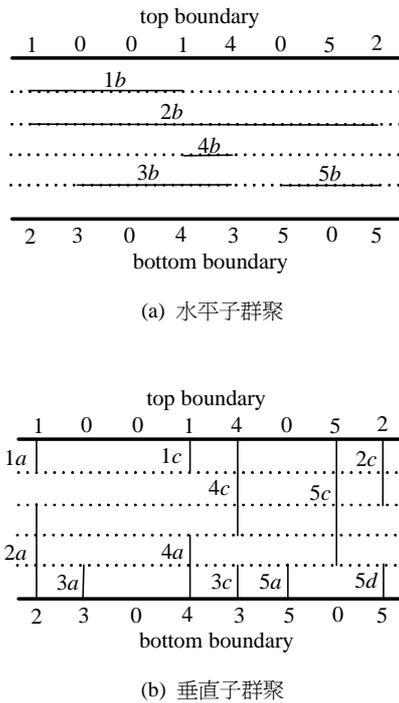


圖 4. 將圖 3 唯一之群聚切割成一水平子群聚及一垂直子群聚

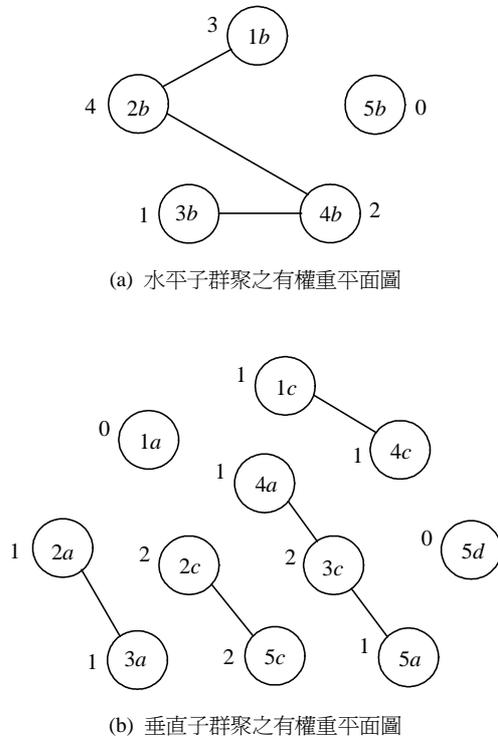


圖 5. 針對圖 4(a) 及 4(b) 之水平及垂直子群聚分別建立有權重平面圖

dependent set) 並將其所含網列線段指定至佈局層之一，再將剩餘網列線段擺放至另一佈局層，以解決群聚內之線段佈局層重新指定問題。圖 6 為圖 3 經過佈局層重新指定後之最終繞線結果。

在圖 6 繞線區域中以實線表示的網列線段擺放在第一佈局層，而以虛線表示的網列線段則擺放在第二佈局層，因此可看出經過佈局層重新指定後之通道繞線結果危險區域的數目為 2。佈局層重新指定演算法步驟一及步驟二的時間複雜度為 $O(n^2 + m^2)$ ，其中 n 及 m 分別代表通道繞線中垂直線段及水平線段數目；步驟三及步驟四的時間複雜度亦為 $O(n^2 + m^2)$ ；因此佈局層重新指定演算法的時間複雜度為 $O(n^2 + m^2)$ 。然而因軌道重新排列演算法的時間複雜度為 $O(d^2)$ ，一般而言， d 值遠小於 n 或 m 值，故整個演算法的時間複雜度則為 $O(n^2 + m^2)$ 。

四、實驗結果

整個後置通道繞線演算法以 C 語言完成並以 PC 作為測試平台 (CPU: Pentium M1.4G with 256MB RAM; OS: Window 2000)。測試電路規格 (亦即初始繞線結果) 如表 1 所示。所有電路初始繞線結果皆取自於 [12]。表 2 顯示後置通道繞線演算法在表 1 例子中之執行效能。平均來說，所提之方法可減少 11.4% 的危險區域。

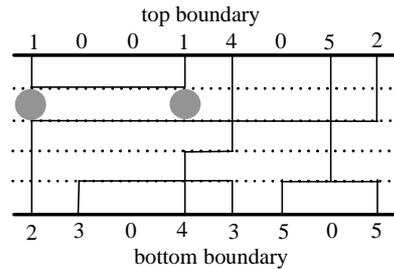


圖 6. 圖 3 經過佈局層重新指定後之通道繞線結果

表 1. 測試電路規格

Circuit	No. of tracks	No. of Columns	No. of nets
Example1	12	35	21
Example3a	15	45	30
Example3b	17	61	47
Example3c	18	79	54
Example5	20	119	63
Deutsch	19	174	72

表 2. 危險區域數目的比較

Circuit	後置演算法執行前	後置演算法執行後	減少比率 (%)	演算法執行時間 (sec.)
Example1	253	221	12.6	0.2
Example3a	664	608	8.4	0.7
Example3b	557	517	7.2	1.2
Example3c	825	779	5.6	1.4
Example5	949	778	18.0	4.9
Deutsch	2721	2256	17.0	8.7

五、結論

我們已提出一個可減少危險區域的後置通道繞線演算法。演算法中藉由軌道重新排列及佈局層重新指定兩個階段來有系統地減少危險區域的數目。實驗結果證實此後置演算法確實能有效降低危險區域的數目。未來我們打算將此演算法擴展應用至多層通道繞線模式中。

誌謝

在此感謝行政院國家科學委員會提供本研究計畫(計畫編號: NSC91-2215-E-212-005) 之經費補助。

參考文獻

1. 林增文、陳漢宗、程仲勝 (民 91), 考慮電磁相容之通道繞線器, 第四屆台灣電磁相容研討會, 台北。
2. Bakoglu, H. B. (1990) *Circuits, Interconnections, and Packaging for VLSI*, 281-303. Addison-Wesley, New York, NY.
3. Chen, H. H. and C. K. Wong (1992) Wiring and crosstalk avoidance in multi-chip module design. 13th Custom Integrated Circuits Conference, Paris.
4. Cherng, J. S. (2003) An optimal algorithm for minimizing

crosstalk for strait-type river routing. 5th Taiwan EMC Conference, Taipei.

5. Cherng, J. S., S. J. Chen and J. M. Ho (1997) Crosstalk minimization in river routing. 8th VLSI Design/CAD Symposium, Taichung.
6. Gao, T. and C. L. Liu (1993) Minimum crosstalk channel routing. 11th International Conference on Computer Aided Design, Santa Clara.
7. Gao, T. and C. L. Liu (1994) Minimum crosstalk switchbox routing. 12th International Conference on Computer Aided Design, San Jose.
8. Jhang, K. S., S. Ha and C. S. Jhon (1996) COP: A crosstalk optimizer for gridded channel routing. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 15(4), 424-429.
9. Kuo, S. Y. (1993) YOR: A yield-optimizing routing algorithm by minimizing critical areas and vias. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 12(9), 1303-1311.
10. Onozawa, A., K. Chaudhary and E. S. Kuh (1995) Performance driven spacing algorithms using attractive and repulsive constraints for submicron LSI's. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14(6), 707-719.
11. Thakur, S., K. Y. Chao and D. F. Wang (1995) An optimal layer assignment algorithm for minimizing crosstalk for three layer VHV channel routing. International Symposium on Circuits and Systems, Seattle.
12. Yoshimura, T. and E. S. Kuh (1982) Efficient algorithms for channel routing. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1(1), 180-190.

收件: 93.07.09 修正: 93.08.11 接受: 93.09.21